



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|-----------------|-------------|----------------------|---------------------|------------------|
| 09/717,587      | 11/21/2000  | Christopher G. Kaler | 777.338US1          | 8807             |

23377 7590 03/23/2004

WOODCOCK WASHBURN LLP  
ONE LIBERTY PLACE, 46TH FLOOR  
1650 MARKET STREET  
PHILADELPHIA, PA 19103

EXAMINER

VU, TUAN A

| ART UNIT | PAPER NUMBER |
|----------|--------------|
|----------|--------------|

2124

DATE MAILED: 03/23/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

**Office Action Summary**

Application No.

09/717,587

Applicant(s)

KALER ET AL.

Examiner

Tuan A Vu

Art Unit

2124

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
  - If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
  - If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
  - Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 15 January 2004.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-34 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-34 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 15 January 2004 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_.
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date \_\_\_\_\_.
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_.

### DETAILED ACTION

1. This action is responsive to the Applicant's response filed 1/15/2004.

As indicated in Applicant's response, claims 1 and 22 have been amended. Claims 1-34 are pending in the office action. The papers filed along with response, e.g. amended drawings, are also entered.

#### *Double Patenting*

2. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and, *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

3. Claims 1-34 are rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-7, 9-24 of U.S. Patent No. 09/717645 (hereinafter '645) in view of Leblang et al., USPN: 5,649,200 (i.e. Leblang). Although the conflicting claims are not identical, they are not patentably distinct from each other because the instant claims represent obvious variations of the invention recited in the claims of the '645 application. The following are but a few examples of such conflicts.

As per instant claims 1 and 22, '645 claims 2 and 10 also recite setting a start time with a value representing the current time in the data structure; but these '645 claims do not disclose creating a link content data structure; setting a link set reference field referring to a link set data

Art Unit: 2124

structure corresponding to set of associated project management objects; and setting an object reference field to refer to the project management object as recited in the instant claims. In an analogous method to control objects versioning, Leblang discloses the setting and creating of link set and link data structure or object reference field as claimed (col. 9, lines 8-56; *configuration record* – Fig. 23; *derived object 500* – Fig. 21; col. 32, line 55 to col. 39; Fig. 20; *VOB* -Fig. 22; link 530 - Fig. 23). It would have been obvious for one of ordinary skill in the art at the time the invention was made to add the creation of content data structure and reference link set as claimed to the '645 invention because with such link set and data structure, the method of updating database by '645 would be more enhanced and fault-free when separate data structures are created and set to support users update instances and thereby keep the database from being overwhelmed by simultaneous update operations with potential contention issues.

**As per instant claims 3 and 24**, '645 claims 6, 14 also recite such data structure being a row in a database.

**As per instant claims 6 and 27**, '645 claims 2 and 10 also recite setting an end time field but do not recite some limitations for which the teachings by Leblang as set forth from above would have rendered obvious, such limitations being receiving a link set identifier, a reference to the managed object, and locating a link content data structure.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

#### ***Claim Rejections - 35 USC § 102***

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

Art Unit: 2124

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 16, 17, 20 and 21 are rejected under 35 U.S.C. 102(b) as being anticipated by Leblang et al., USPN: 5,649,200 ( hereinafter Leblang).

**As per claim 16**, Leblang discloses a processor and readable medium; operating environment for the readable medium; and project management system to maintain version of associations between management objects (e.g. Fig. 1-2; Fig. 6-7, Figs 15-23; col. 2, lines 23-42; col. 9, lines 8-56 – Note: configuration record implicitly discloses persistent storage of structure associating project objects; and records created being built and kept for builds are implied versions made according to the builds when compared to other versions for other builds).

**As per claim 17**, Leblang further discloses that the project management system include a versioned file database (e.g. VOB 20 – Fig. 1; *version tree* – Fig. 3); a project data database to store project data ( e.g. developer's work station/database 102 – Fig. 2; col. 6, lines 21-25; *VOB 102* ( right) – Fig. 6); and project associations database to store associations between project data and versioned files (e.g. *config. record 532* –Fig. 23; col. 6, lines 21-25; private database 22 – Fig. 1; *reuse derived objects* – col. 4, lines 11-21; *date-time* -col. 21, line 46 to col. 23, line 52).

**As per claim 20**, Leblang discloses database for storing both project data and association data (e.g. VOB 102 – Fig. 23).

**As per claim 21**, Leblang discloses a source code repository ( e.g. *VOB 102* ( left) – Fig. 6; Fig. 17).

***Claim Rejections - 35 USC § 103***

Art Unit: 2124

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. Claims 1, 3-6, 8-11, 13-14, 18-19, 22, 24-27, 29-32 and 34 are rejected under 35

U.S.C. 103(a) as being unpatentable over Leblang et al., USPN: 5,649,200 ( hereinafter

Leblang), in view of Eisenberg et al., USPN: 5,890,166 ( hereinafter Eisenberg).

As per claim 1, Leblang discloses a computerized method for adding (e.g. *merge, check-in* – Fig. 13-15; col. 27, lines 50-67) an association of a project management object ( hereinafter PMO) to a set of associated project management objects, the method comprising:

creating a link content data structure (e.g. *view ...build ... Release* – col. 9, lines 8-56; *configuration record* – Fig. 23);

setting a link set reference field in said link content data structure to a value that refers to a link set data structure corresponding to the set of associated PMOs (e.g. entry 532, link 530; Fig. 20; *VOB* -Fig. 22; link 530 - Fig. 23 – Note: link set reference field, an entry stored link content data structure, i.e. *configuration record*, has a value calling for link 530 – a link set data structure corresponding to more than on objects, such *derived objects 500* from a VOB database of linked derived objects being equivalent to set of PMOs);

setting an object reference field in said link content data structure to refer to the project management object (e.g. an field calling for link 530 - Fig. 23).

But Leblang does not specify setting a start time field in the link content data structure to a value representing the current. However, Leblang teaches associating a time stamp with the

Art Unit: 2124

referred to objects for update (e.g. Fig. 8-9; col. 11, lines 44-51; col. 16, line 49-55). Besides, the associating of time stamp to version control and informing on life scope of versioned objects was a well-known concept at the time of the invention. Further, in a method to control versioning analogous to Leblang's method, Eisenberg discloses setting a start time field in the link content data structure to a value representing the current time (e.g. col. 15, lines 17-20, 27-29) and setting an end time field in the link content data structure to a value representing a most recent version of the object (e.g. e.g. col. 5, lines 39-41). It would have been obvious for one of ordinary skill in the art at the time the invention was made to provide the techniques of setting the start and end time as suggested by Eisenberg to Leblang's time stamp attribute because the recording of time variance or time elapsed related to database update operations enable better understanding of real-world interdependencies of versioned objects, their most current versions, and the dynamic state of their being updated by more than one operators over time ( see Eisenberg: col. 14, line 1 to col. 17, line 67).

**As per claim 3**, Leblang teaches that the link content data structure is a row ( e.g. *record 514* – Fig. 20; Fig. 21 – Note: one entry for the record in the derived object table is equivalent to a row).

**As per claim 4**, Leblang does not explicitly specify that accessing the derived objects for version update, i.e. link set data structure, means referring to a row in a database; but teaches a table or record of the object to modify ( e.g. *VERSION OBJECT* , *additional fields* – Fig. 9), hence implicitly teaches row to such record.

**As per claim 5**, Leblang disclose include/header files or object code, meta-data and program source, release notes and scripts (e.g. col. 28, line 33 to col. 29, line 31; Fig. 17; record 532 – Fig. 23; *rlsnotes 208* – Fig. 6); hence discloses PMOs as claimed.

**As per claim 6**, Leblang discloses a computerized method for removing (e.g. *merge*, *check-in* – Fig. 13-15) an association of a project management object ( hereinafter PMO) from a set of associated project management objects, the method comprising:

receiving an identifier for a link set corresponding to the set of associated PMOs (e.g. link 530, *derived object 500* - Fig. 22, 23);

receiving a reference to the PMO (e.g. col. 32, line 55 to col. 39; link 530 - Fig. 23);

locating a link content data structure containing the reference to the PMO (e.g. record 514 – Fig. 20; *config rec 530*, link 530 – Fig. 21).

But Leblang fails to specify setting an end time field in the link content data structure to a value representing the current time. But this limitation has been addressed in claim 1 above using Eisenberg's teachings, hence is rejected herein likewise.

**As per claims 8-9**, refer to rejection of claims 4 and 3, respectively.

**As per claim 10**, see claim 5.

**As per claim 11**, Leblang discloses a method for retrieving a set of project management objects associated (e.g. col. 2, lines 23-52 ) with a source program management object ( PMO), the method comprising:

receiving a reference to the source program management object and a time value (e.g. col. 3, lines 10-18; link 530 - Fig. 23; col. 11, lines 17-60; *date-time* -col. 21, line 46 to col. 23,



Art Unit: 2124

line 52 – Note: getting a reference to a derived object and version-selector to select version of object for build and update with associated time stamp is equivalent to receiving a time value);

querying a set of link content data structures to create a set of valid link content data structures (e.g. col. 4, lines 11-27; col. 11, lines 17-60 – Note: retrieving configuration records to point -- set of link content data structure – entry 532, or pointer 530, Fig. 23 -- to a set of derived objects in accordance with build/release requirements is equivalent to querying a set of link content data structures to set a valid set thereof), wherein each link content data structure contains reference to the source PMO (e.g. *file name, versions of sources, dependencies* - Fig. 20; *derived object 500* – Fig. 23 );

creating a set of source link set references comprising the link set reference contained in the set of valid link content data structures (e.g. entries 532 of *config record*, link 530 to *derived object 500* – Fig. 23 – Note: source link set references are entries in the *Config Records* among identified valid *Config Records* or link content data structures; and using links 530 or link set reference, to correspond to said source link set references --e.g. entries 532 in *Config Record*-- reads on creating of set of source link set references with link set reference contained the set of valid link content data structures);

querying the set of link content data structures to create a set of matching PMOs for each source link set reference (e.g. col. 2, line 23-34; col. 10, line 53 to col. 11, line 59 – Note: for each reference or entries set by the *config record* or bill-of-materials, selecting the correct set of derived objects is equivalent to querying as claimed ), wherein each matching PMO has a link set reference equal to the source link set reference (e.g. entries 532 of *config record*, link 530 to *derived object 500* – Fig. 23).

But Leblang does not disclose that each valid link content data structure contains a start time less or equal to the time value and an end time greater or equal to the time value; nor does Leblang specify that the PMO has a start time and end time  $\leq$  or  $\geq$  such time value. But, according to the method of Eisenberg as mentioned in claim 1, the use of time variants as field of the database record so to determine which version has been superseded by another version discloses the limitation that the time value as claimed has to fall between such start and end time (col. 14, line 38 to col. 15, line 49). Hence, in view of the time stamp teachings by Leblang and the added features by Eisenberg to address the varying state of a timed version of objects, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify Leblang's method for storing objects ( i.e. link content data structure, matching PMO) so as to include the start and end time fields as suggested by Eisenberg because this would enhance the version checking and promoting of version based on knowledge time variants fields as shown by Eisenberg.

**As per claim 13**, see claim 5.

**As per claim 14**, Leblang discloses a data structure comprising:

a first field comprising a reference to a link set data structure corresponding to a set of associated PMOs (e.g. entries 523, link 530 - Fig. 23 );

But Leblang does not specify a second field comprising a start time, a third field comprising a end time, wherein the second and third field define a range of time that the target PMO is associated with the set of associated PMOs. But the limitation of having 2 such fields to use as range for determining time of the target PMO has been addressed in claim 11 above.

Art Unit: 2124

**As per claims 18 and 19**, with reference to claim 17, Leblang fails to specify that the project data and association data database is a relational database. But Eisenberg, in a similar system to version control objects, suggests relational database for the preferred embodiment (e.g. col. 1, lines 41-53). It would have been obvious for one of ordinary skill in the art at the time the invention was made to implement Leblang database system using the suggested relational type DB by Eisenberg, since this type of database is the most well-known and used by many enterprises and would make it more efficient and friendly for the storing, normalizing and querying of records or data.

**As per claim 22**, this is a computer medium claim of corresponding claim 1; hence is rejected with the same rationale used therein.

**As per claims 24-26**, these claims correspond to claims 3-5; and are rejected likewise, respectively.

**As per claim 27**, this is a computer medium claim of corresponding claim 6; hence is rejected with the same rationale used therein.

**As per claims 29-31**, refer to claims 3-5, respectively, for corresponding rejections.

**As per claim 32**, this is a computer medium claim of corresponding claim 11; hence is rejected with the same rationale used therein.

**As per claim 34**, see claim 5.

8. Claims 2, 7, 12, 15, 23, 28, and 33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Leblang et al., USPN: 5,649,200, and Eisenberg et al., USPN: 5,890,166, as applied to claims 1, 6, 11, 14, 22, 27, 32 from above, in view of Reed et al., USPN: 5,862,325 (hereinafter Reed).

**As per claim 2**, Leblang only teaches link being hard pathname but does not specify object reference field being an URL. However, Leblang teaches hyperlinks to reach versioned data or database (e.g. col. 16, line 57 to col. 17, line 29). Reed, in a communication scheme to enable update of database versioned data analogous to Leblang's version control method using system hard links, discloses use of URL to reach out for version instances of objects (e.g. *http://company.com/commobject3481.cos* -- col. 91, line 27 to col. 92, line 9; Fig. 2). It would have been obvious for one of ordinary skill in the art at the time the invention was made to provide such URL to locate version database as suggested by Reed, and apply this to Leblang linking method because database accessible across the internet would alleviate extraneous storage resources of the local system or environment under which Leblang's method operates for version update.

**As per claim 7**, this claim corresponds to claim 2 above, hence is rejected herein using the same rationale as set forth therein.

**As per claim 12**, this claim corresponds to claim 2 above, hence is rejected herein using the same rationale as set forth therein.

**As per claim 15**, see claim 2.

**As per claims 23 and 28**, see claim 2.

**As per claim 33**, see claim 2 for corresponding rejection.

### ***Response to Arguments***

9. Applicant's arguments filed 1/15/2004 have been fully considered but they are not persuasive. Following are the Examiner's remarks therefor.

Art Unit: 2124

(A) **As per claims 16-21**, Applicants have submitted that Leblang ‘contradicts a view that directory objects ...could be analogized to ... can not be associated using a directory ... do not associate the elements.’ ( Appl. Rmrks, pg. 14, 2<sup>nd</sup> para); and that “the directories in Leblang ... not analogous to associations in the claimed invention” (Appl. Rmrks, pg. 14, 3<sup>rd</sup> para). For the sake of argument, even if such directories concept were used to support versioned associations of project management objects (PMOs) as brought up by Applicants, it is noted that those directories are versioned and represent a structure having its own version and depicting association between PMOs as a hierarchy linking those PMOs together and do read on versioned associations ( see Leblang: col. 3, lines 31-40). However, the rejection cites a different part of Leblang to read on the claimed feature in question here by Applicants; rendering the point thus raised a moot point.

**As per claims 1-15 and 22-24:**

(B) As per claim 1, Applicants have submitted that Leblang’s ‘view’ and ‘configuration record’ or ‘derived object’ do not constitute what is claimed as a link content data structure. First, it is argued that a ‘view’ is open-ended and there is no-need to add files to a view; or that no addition or creation of link content data structure can be effected therein (Appl. Rmrks, pg. 16, bottom para ). The claim only recites the steps of creating a link content data structure, setting a field therein with a value referring to another link set structure in another set of PMO and more setting of fields. There is no adding explicitly recited in the claimed, i.e. not a single action depicting a clear occurrence of adding any element to another so to tie with what has been proclaimed by the preamble, if not saying that such lack of connection suggests a potential non-statutory status of an invention. The ‘view’ in Leblang at least is a place wherein objects are

Art Unit: 2124

added in order to represent what is prescribed by the requirements of a bill of material or a configuration designed for a build or release because it is inherent that the so-called view be an user interface to which additional modifications are to be performed by the developer. And as claimed, there is not sufficient evidence that any step of adding is to be a required feature to address.

(C) Further, Applicants have submitted that a 'configuration record' by Leblang does not teach modification, and that 'derived objects' do not change over time and that there is no teaching of 'a new association of a project ... field of the data structure)' (Appl. Rmrks, pg. 17, 2<sup>nd</sup> and 3<sup>rd</sup> para). As recited, the claim recites a method comprising steps creating, setting, setting, and setting. The claim does not specify the step of creating with details as to point out an order by which, the circumstances under which, or an event based upon which such creation is performed. The configuration records by Leblang have to be created otherwise they would not be there for the project developer to retrieve; and such creation reads on the claim, in view of the observation from above. Further, the claim does not recite any step of modifying, hence the cited portions of Leblang to set link and fields in the configuration objects still reads on the steps of setting. Here again, there is not explicit sequence in the claim for enforcing that a modification is taking place or that a setting is subsequent to any particular steps ( Note: the claim recites a method comprising: ); hence the point about derived objects susceptible to be changed over time amounts to no limitation to be addressed. The cited portions show a structure in which field pointing to another PMO and these citations are fulfilling the required steps as recited.

Art Unit: 2124

(D) Applicants have submitted that a 'derived object' by Leblang cannot be analogized to either a link content data structure or a link set data structure and that "upon the addition of a file ... is created ... contains a reference to the link set data structure" (Appl. Rmrks, pg. 17, bottom; pg. 18, top para). As mentioned above in section C, there is not explicit specification by the claim to enforce an order for performing the step of creating or a required feature of adding. The claim does not recite that a link content data structure must be created upon the addition of a new object to an existing association; nor is there explicit requirement that a link set data structure be created upon the addition of a file to the association represented by a derived object. Examiner has clarified any issue by pointing the Applicants to more appropriate set of citations. The rejection herein points out what the link content data structure is, what constitutes a link set reference field referring thereto, and a value referring such link set data structure corresponding to PMOs; and the arguments raised by Applicants would become moot as a result of such set of cited portions used in the rejection.

(E) As for arguments raised against Leblang's cited teaching on *full pathname* and *a\_slink* (Appl. Rmrks, pg. 18, middle para), Examiner, as mentioned above, has redirected Applicants' attention to where in the Leblang's reference the limitations as mentioned above are to be matched against; i.e. link content data structure (configuration record); link set reference field/value (entry in such record); link set data structure (a structure representing a derived objects); a PMO ( any derived objects, or script or file in the VOB).

(F) As per arguments against claims 2-5; 6-10 (Appl. Rmrks, pg. 18, bottom; pg. 19), these arguments ( e.g. adding new structure, derived objects do not change over time, full pathname), incorporate the arguments already addressed in section A-E; hence are referred back thereto.

Art Unit: 2124

(G) As per claims 11-15, Applicants have submitted that a set of valid content data structures and a set of PMO are separate elements (Appl. Rmrks, pg. 21, top). The rejection has pointed out that valid content data structures are identified configuration record and what is interpreted as the limitation recited as link set reference structure. Indeed, this latter limitation is a pointing structure (pointer 520, Fig. 23) embodying the source link set references or entries in the content data structure--such structure being a configuration record-- and is used to point to a valid set of PMOs.

Further, it is argued that derived objects and version-selector do not fulfill "finding link content data structure with the time value" and that Leblang does not fulfill the limitation of valid link content data structure used to find program management objects (Appl. Rmrks, pg. 21, 2<sup>nd</sup> para, 3<sup>rd</sup> para ). The one limitation to address is a time value as to enable a comparison via version-matching in order to validate time-wise a set of link content data structures aimed at building a release. The rejection using the combination Leblang/Eisenberg has fulfilled this time-related point (see *date-time* -col. 21, line 46 to col. 23, line 52). Leblang provides associating the PMO with a version-matching tool. This validation based on time stamp comparing, reads, at least implicitly, on querying of valid link set content data structure -- all of such structures are pointing structures associating PMO related to some time stamp information. Eisenberg provides the teaching on start time and end time. The combination as set forth provides both the finding of link content structure pointing to versioned objects associated with a time stamp and a time stamp for a start and end time. Since the configuration record points to PMO, the fact of having a structure referring to the PMO, such pointing structure (i.e. link content data structure) found among many *config records* read on querying a set of link content



Art Unit: 2124

data structures which are made valid when used to identify PMOs being the derived objects structures; and the rejection has shown that. The point raised concerning reuse would be moot.

Further, as for claim 14, Applicants have argued that no time limit or version in the association between link content data structure and PMOs are addressed in the prior art used (Appl. Rmrks, pg. 21, 4<sup>th</sup> para). This argument has been carefully addressed in section G above and in the rejection.

For the above reasons, the claims stand rejected as set forth in the present rejection.

***Conclusion***

10. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (703)305-7207. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

Art Unit: 2124

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (703)305-9662.

**Any response to this action should be mailed to:**

Commissioner of Patents and Trademarks

Washington, D.C. 20231

**or faxed to:**

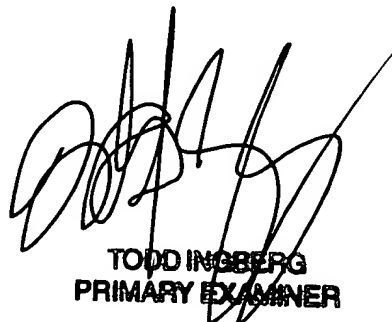
(703) 872-9306 ( for formal communications intended for entry)

**or:** (703) 746-8734 ( for informal or draft communications, please consult Examiner before using this number)

Hand-delivered responses should be brought to Crystal Park II, 2121 Crystal Drive, Arlington. VA. , 22202. 4<sup>th</sup> Floor( Receptionist).

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

VAT  
March 11, 2004



TODD INBERG  
PRIMARY EXAMINER